

PATENT COOPERATION TREATY

PCT

INTERNATIONAL PRELIMINARY REPORT ON PATENTABILITY



(Chapter II of the Patent Cooperation Treaty)

(PCT Article 36 and Rule 70)

REC'D 01 DEC 2005

WIPO

PCT

Applicant's or agent's file reference Rapid Dev PCT		FOR FURTHER ACTION		See Form PCT/PEA/416
International application No. PCT/GB2004/004702		International filing date (day/month/year) 08.11.2004	Priority date (day/month/year) 06.11.2003	
International Patent Classification (IPC) or national classification and IPC G06F9/44				
Applicant INTUWAVE LIMITED et al.				
<p>1. This report is the international preliminary examination report, established by this International Preliminary Examining Authority under Article 35 and transmitted to the applicant according to Article 36.</p> <p>2. This REPORT consists of a total of 11 sheets, including this cover sheet.</p> <p>3. This report is also accompanied by ANNEXES, comprising:</p> <p>a. <input checked="" type="checkbox"/> sent to the applicant and to the International Bureau) a total of 10 sheets, as follows:</p> <p><input type="checkbox"/> sheets of the description, claims and/or drawings which have been amended and are the basis of this report and/or sheets containing rectifications authorized by this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions).</p> <p><input checked="" type="checkbox"/> sheets which supersede earlier sheets, but which this Authority considers contain an amendment that goes beyond the disclosure in the international application as filed, as indicated in item 4 of Box No. I and the Supplemental Box.</p> <p>b. <input type="checkbox"/> (sent to the International Bureau only) a total of (indicate type and number of electronic carrier(s)) , containing a sequence listing and/or tables related thereto, in computer readable form only, as indicated in the Supplemental Box Relating to Sequence Listing (see Section 802 of the Administrative Instructions).</p>				
<p>4. This report contains indications relating to the following items:</p> <p><input checked="" type="checkbox"/> Box No. I Basis of the opinion</p> <p><input checked="" type="checkbox"/> Box No. II Priority</p> <p><input type="checkbox"/> Box No. III Non-establishment of opinion with regard to novelty, inventive step and industrial applicability</p> <p><input type="checkbox"/> Box No. IV Lack of unity of invention</p> <p><input checked="" type="checkbox"/> Box No. V Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement</p> <p><input checked="" type="checkbox"/> Box No. VI Certain documents cited</p> <p><input type="checkbox"/> Box No. VII Certain defects in the international application</p> <p><input type="checkbox"/> Box No. VIII Certain observations on the international application</p>				
Date of submission of the demand 06.09.2005		Date of completion of this report 30.11.2005		
Name and mailing address of the international preliminary examining authority:  European Patent Office D-80298 Munich Tel. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465		Authorized Officer Bösch, M Telephone No. +49 89 2399-7523 		

**INTERNATIONAL PRELIMINARY REPORT
ON PATENTABILITY**

International application No.
PCT/GB2004/004702

Box No. I Basis of the report

1. With regard to the **language**, this report is based on the international application in the language in which it was filed, unless otherwise indicated under this item.
- ☐ This report is based on translations from the original language into the following language , which is the language of a translation furnished for the purposes of:
- ☐ international search (under Rules 12.3 and 23.1(b))
 - ☐ publication of the international application (under Rule 12.4)
 - ☐ international preliminary examination (under Rules 55.2 and/or 55.3)
2. With regard to the **elements*** of the international application, this report is based on *(replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to this report)*:

Description, Pages

1, 2, 9-46	as originally filed
3-8	filed with telefax on 17.11.2005

Claims, Numbers

1-30	filed with telefax on 17.11.2005
------	----------------------------------

Drawings, Sheets

1/2, 2/2	as originally filed
----------	---------------------

- ☐ a sequence listing and/or any related table(s) - see Supplemental Box Relating to Sequence Listing

3. ☒ The amendments have resulted in the cancellation of:
- ☐ the description, pages
 - ☒ the claims, Nos. 31-34
 - ☐ the drawings, sheets/figs
 - ☐ the sequence listing (*specify*):
 - ☐ any table(s) related to sequence listing (*specify*):
4. ☒ This report has been established as if (some of) the amendments annexed to this report and listed below had not been made, since they have been considered to go beyond the disclosure as filed, as indicated in the Supplemental Box (Rule 70.2(c)).
- ☒ the description, pages 3-8
 - ☒ the claims, Nos. 1-30
 - ☐ the drawings, sheets/figs
 - ☐ the sequence listing (*specify*):
 - ☐ any table(s) related to sequence listing (*specify*):

* If item 4 applies, some or all of these sheets may be marked "superseded."

**INTERNATIONAL PRELIMINARY REPORT
ON PATENTABILITY**

International application No.
PCT/GB2004/004702

Box No. II Priority

1. ☐ This report has been established as if no priority had been claimed due to the failure to furnish within the prescribed time limit the requested:
- ☐ copy of the earlier application whose priority has been claimed (Rule 66.7(a)).
 - ☐ translation of the earlier application whose priority has been claimed (Rule 66.7(b)).
2. ☐ This report has been established as if no priority had been claimed due to the fact that the priority claim has been found invalid (Rule 64.1). Thus for the purposes of this report, the international filing date indicated above is considered to be the relevant date.
3. Additional observations, if necessary:
- see separate sheet**

Box No. V Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Statement

Novelty (N)	Yes: Claims	2,3,22,24-26
	No: Claims	1,4-21,23,27-34
Inventive step (IS)	Yes: Claims	
	No: Claims	1-34
Industrial applicability (IA)	Yes: Claims	1-34
	No: Claims	

2. Citations and explanations (Rule 70.7):

see separate sheet

Box No. VI Certain documents cited

1. Certain published documents (Rule 70.10)

and / or

2. Non-written disclosures (Rule 70.9)

see separate sheet

Concerning Section I,
Basis of the report

- 1 Since Claim 26 filed with the telefax of 17.11.05 is identical to Claim 34 as originally filed, it is assumed that it was not intended to maintain Claim 34 in the set of claims filed by this telefax.
- 2 The amendments filed with the telefax of 17.11.05 introduce subject-matter which extends beyond the content of the application as filed, contrary to Article 34(2)(b) PCT. The amendments concerned are the following in **Claim 1**:
 - 2.1 Firstly, with this telefax the feature "a computer remote from the mobile device calls, over one of the network connections the networked application communicates with the enterprise server, modular software elements" has been introduced.
 - 2.2 However, according to the application as originally filed, two separate sets of communication links are provided for the application (see page 11, lines 6-11 and Claim 2) and for the remote control link (see e.g. Claim 21), and there is nowhere mentioned the claimed use of the same connection, as handled by the application to be developed, for the remote control as well.
 - 2.3 Secondly, the newly incorporated combination of:
 - a) a developer using a remote computer to call modular software elements running on the device, and
 - b) causing elements on the device to be combined using a scripting engine,was not disclosed in the application as originally filed, which discloses them as alternatives.

In particular, in the application as originally filed it was disclosed to either remotely control the application on the wireless device based on a command line program interpreting the commands received from a remote computer, the commands being in a high level language of one of a command line interface, a script language, a compiled language, or to download a script (e.g. by using the command line interface) and executing the

script with a script engine on the device.

- 2.4 Thirdly, the step of "(c) the developer exploring how different elements respond to inputs by repeating steps a) and b)" is not disclosed in the application as originally filed.

By contrast, page 18, line 19 only supports the repetition of script execution and how the combined elements act, but neither the claimed repetition of a command input and a script execution in combination nor the response of selected elements checked by scripts, and Claim 11 as originally filed only discloses the exploration of certain elements from a command prompt at the remote computer.

- 3 Due to the great number of amendments in Claim 1 which do not meet the requirements of Article 34(2)(b) PCT, an examination about novelty and inventive step concerning this claim and those of the dependent claims adapted to this new claim 1 does not make sense at the moment.
- 4 **Therefore, the examination about novelty and inventive step in section V below has been made on the basis of Claims 1-34 as originally filed**, and the pages of the description filed with the telefax on 17-11-05, which have been adapted to these new claims, are also neglected.

5 **Further remarks concerning Claim 1 filed with telefax on 17.11.05:**

- 5.1 In addition, contrary to the requirements of Rule 6.3 PCT, Claim 1 as filed with telefax on 17.11.05 does neither indicate the technical features which are known to the prior art nor the additional technical features for which in combination with these known features protection is desired.

By contrast, this claim is related to a method of rapid software development defined exclusively by method steps to be done by a developer (i.e. a developer using ..., causing ..., exploring ...), which cannot be regarded as technical features defining a technical method.

So, apparently the method of this claim is *prima facie* not concerned with a technical problem and is not defined by technical features in terms for which protection is sought (see PCT International Search and Preliminary Examination Guidelines of 25.3.04, chapter 9.04).

- 5.2 If this claim 1 was reformulated towards a method comprising steps which are carried out by a technical entity like e.g. the remote computer and/or wireless mobile device (e.g. calling/execution of software elements), the Applicant should note that according to the common practice in the EPO, only the technical features of such a method could contribute to inventive step of the claimed subject-matter.

To put it in a nutshell, if all technical features of a claim are already known from prior art, and there is only a non-technical feature which is not disclosed or rendered obvious, the claimed subject-matter would not involve an inventive step in the sense of Article 33(3) PCT.

In this particular case this means that if the underlying technical development architecture (elements with standard interface encapsulating the device functionalities, executing on the device and being under control of an interpreter) is disclosed in the prior art, the use of the architecture (e.g. developer makes iterations based on modified scripts) is regarded as non-technical feature which does not provide a technical effect and hence does not render the claimed subject-matter inventive.

- 5.3 Finally, according to Rule 6.2 b) PCT the features cited in the claims are supplied with reference signs in parenthesis where elements of the drawings are shown.

However, the bracketed feature "(that is not specific to the mobile device)" of Claim 1 filed with the telefax on 17.11.05 is not a reference sign. Consequently, it should be separated by another type of punctuation mark (cf. analogous Rule 11.13 l) und 11.13 m) PCT), because it is not clear whether this feature is limiting or not since references signs usually are not regarded as limiting a claim.

- 6 In the sections below, reference is made to the following documents, which are cited

in the International Search Report:

- D1: WO 03/036470 A (INTUWAVE LIMITED; SPOONER, DAVID) 1 May 2003 (2003-05-01)
D2: WO 03/065654 A (KONINKLIJKE PHILIPS ELECTRONICS N.V; MELPIGNANO, DIEGO) 7 August 2003 (2003-08-07)
D3: WO 97/35254 A (MASSACHUSETTS INSTITUTE OF TECHNOLOGY) 25 September 1997 (1997-09-25)
D4: EP-A-1 233 635 (MICROSOFT CORPORATION) 21 August 2002 (2002-08-21)
D5: WO 2004/088508 A (INTUWAVE LIMITED; KARDASH, ANATOLY) 14 October 2004 (2004-10-14)

Concerning Section VI.
Certain documents cited:

Certain published documents (see Rule 70.10 PCT)

Application No Patent No	Publication date (day/month/year)	Filing date (day/month/year)	Priority date (valid claim) (day/month/year)
D5=WO 2004/088508	14.10.04	31.3.04	4.4.03

Concerning Section II.
Priority of the international application

The document D5 indicated in the International Search Report as a P-document is not (see Rule 64.3 PCT) to be regarded as state of the art according to Articles 33(2) and (3) PCT, as the date of priority claimed can be allowed for the relevant parts of the present application (see Rule 64.1 b) ii) PCT). However, it may become of high relevance when entering into the national phase (e.g. see Article 54(3) EPC).

Concerning Section V.
Reasoned statement with regard to novelty, inventive step or industrial applicability

- 1 The subject-matter of independent **Claims 1 and 31** as originally filed is not novel (see Article 33 (2) PCT) in the light of prior art document D1 disclosing a method of

rapid software development for a wireless mobile device (see D1, title), comprising the step of calling modular software elements (see D1, page 10, lines 20-24), that each

(i) encapsulate functionality required by the wireless mobile device (see D1, page 5, line 30-page 6, line 6),

(ii) share a standard interface structure (see D1, page 11, lines 1-16, figure 3), and

(iii) execute on the device, under control of a high level language program, for instance a command line interface (see D1, page 11, line 24-30, page 13, lines 26-30, page 16, lines 1-7, 16-21).

- 2 Furthermore, it is pointed out that should the Applicant dispute the aforementioned novelty objection based on minor differences of interpretation between some of the features of these independent claims and the features of D1, the claimed subject-matter would still not seem to involve an inventive step (see Article 33(3) PCT) in the light of the above cited document D1, because there appears to be no feature having an underlying technical effect or solving a technical problem in both independent claims.
- 3 It is furthermore pointed out that even if the above novelty objection were overcome by amendments of a minor nature, the subject-matter of independent Claims 1 and 31 would still not seem to involve an inventive step (see Article 33(3) PCT), considering that D1 aims at the same object and discloses same type of solution as the present application.
- 4 The subject-matter of the software application **Claim 27** as originally filed lacks novelty in the sense of Article 33(2) PCT and an inventive step (Article 33(3) PCT) for the same reasons.
- 5 The dependent **Claims 2-26, 28-30 and 32-34** as originally filed do not appear to contain any additional features which, in combination with the features of any claim to which they refer, meet the requirements of the PCT with respect to novelty and/or inventive step, the reasons therefore are that their features have already been employed for the same purpose in the prior art (**Claims 2, 3, 33**: see D2, page 6, line 28 - page 7, line 2, page 7, lines 22-33, page 11, lines 16-30, figure 3, **Claims 4, 5, 33**: see D1, page 5, line 30 - page 6, line 21, **Claims 6, 33**: see D1, page 4, line 29,

page 16, lines 23-29, **Claims 7, 8, 17, 19, 32, 33**: see D1, page 16, lines 1-7, 16-21, **Claims 9-15, 18, 21, 28, 29, 30, 33**: see D1, page 16, lines 1-7, 16-21, page 20, line 20-page 21, line 21, figure 11, **Claims 16, 33**: see D1, page 11, lines 1-16, **Claims 20, 33**: see D1, page 3, lines 14-17, 28-31, **Claims 22, 33**: see D1, page 5, lines 12f, page 6, lines 3f, see D2, page 6, line 28 - page 7, line 2, page 7, lines 22-33, page 11, lines 16-30, figure 3, **Claims 23, 33**: see D1, figure 6, **Claim 34**: see D1, page 5, lines 17-19), don't have an underlying technical effect, or consist of slight constructional changes which come within the scope of the customary practice followed by persons skilled in the art.

- 6 It is not apparent which part of the application could serve as a basis for a new claim which would be in agreement with the PCT. However, if the Applicant regards some particular matter as new and inventive, an independent claim including such matter should be filed, e.g. when entering the national phase.

Then, the Applicant should also indicate in the letter of reply the difference of the subject-matter of the new claim vis-à-vis the state of the art (novelty, Article 33(2) PCT) and the significance **and underlying technical effect** thereof (inventive step, Article 33(3) PCT) in a very detailed manner!

- 7 Since the claimed subject-matter is already industrially applied or is at least already suggested by combinations of the documents D1 and D2, the requirements of Article 33(4) PCT are fulfilled.

8 **Remarks concerning clarity of the international application:**

- 8.1 The application contains 2 independent claims relating to a method of rapid software development (**Claims 1 and 31** as originally filed). As these claims are of overlapping scope, the requirements of Article 6 PCT and Rule 6.1 (a) PCT regarding conciseness of the claims are not met. In detail, Claim 1 represents just the alternative a) of a high level language as suggested in dependent Claim 32!
- 8.2 From the present formulation of **Claim 27** as originally filed, it is not clear how a software application developed using the method of Claim 1 differs from a software application developed in a traditional manner. This claim could be clarified as follows:

"A software application comprising modular elements, that each (i) encapsulate ...".

- 8.3 In the light of the Rule 5.1(a)(iii) PCT, there must not exist any inconsistencies between the description and the claims, as it may throw doubt on the extent of protection and therefore render the claim unclear in the sense of Article 6 PCT. However, the present formulation of **Claim 33** as originally filed implies that all features of Claim 31 are defined twice, thereby defining a method having two times these features, which apparently is in contradiction to the description.
- 8.4 Similar comments apply to **Claim 17** as originally filed, because the present formulation implies that one of these alternatives can be selected, whereas Claim 1 already comprises the command line interface (first alternative).
- 8.5 The features in the software application **Claims 28-30** as originally filed are formulated as method steps rather than clearly defining the software in terms of its code means, contrary to the requirements of Article 6 PCT.
- 8.6 The **Claims 19 and 23** as originally filed do not meet the requirements of Article 6 PCT in that the matter for which protection is sought is not defined. The claims attempt to define the subject-matter in terms of the result to be achieved which merely amounts to a statement of the underlying problems. The technical features necessary for achieving these results should have been added.
- 8.7 The term "device networking functions" of **Claim 3** as originally filed lacks an antecedent basis and as such renders the scope of the claim unclear. It probably should refer to the "device networking functionality" mentioned before.
- 8.8 Similar comments apply to **Claims 6-9** ("the network element under control of a command line interface") and **Claims 11 and 13** ("the device-based elements controlled by a command line interface") as originally filed.
- 9 **Remarks concerning formal defects in the international application:**
- The independent claims should have been drafted in the proper two-part "characterised" form recommended by Rule 6.3.(b),(i),(ii) PCT, having a preamble that

correctly reflects the nearest prior art, presumably that represented by the above noted D1.

- If necessary, the description should be brought into conformance with any claims newly filed, e.g. when entering the national phase.
- Contrary to the requirements of Rule 5.1 (a),(ii) PCT, the relevant background art disclosed in the documents D1, D2 and D4 is not mentioned in the description, nor are these documents identified therein.
- Reference signs in parentheses should have been inserted in the claims to increase their intelligibility (see Rule 6.2 (b) PCT).
- The attention of the Applicant is finally drawn to the fact that, when entering the national phase, the application may not be amended in such a way that it contains subject-matter which extends beyond the content of the application as filed (see Articles of the national law corresponding to Article 34.2(b) PCT like e.g. Article 123(2) EPC).
- In order to facilitate the examination of the conformity of the amended application with the requirements of these Articles, the Applicant is requested to clearly identify the amendments carried out, no matter whether they concern amendments by addition, replacement or deletion, and to indicate the passages of the application as filed on which these amendments are based (see also Rule 66.8(a) PCT).

Rapid Dev

3

SUMMARY OF THE PRESENT INVENTION

In a first aspect, there is a method of rapid software application development for a wireless mobile device, the application being an enterprise networked application in which the device communicates with an enterprise server over one or more types of network connection; the method comprising the steps of:

- (a) a developer using a standard, high level interface (that is not specific to the mobile device OS) on a computer remote from the mobile device to call, over one of the network connections, modular software elements running on the device, the modular elements each (i) encapsulating functionality required by the wireless mobile device and (ii) executing on the device, under the control of an interpreter of the high level interface;
- (b) the developer causing elements on the device to be combined using a scripting engine running on the device; and
- (c) the developer exploring how different elements respond to inputs by repeating steps (a) and (b).

This approach makes it far easier for a programmer to explore the functioning of the elements – in particular how an element responds to a given input. The kind of visibility of functioning is very difficult to achieve using a conventional development methodology.

One or more modular software elements encapsulate device networking functions. The device networking functionality relates to connectivity over one or more of the following: GPRS, 2G cellular, CDMA, WCDMA, Bluetooth, 802.11, infra-red, IP networking, dial up, modem; HSCSD and EDGE. Previously, being able to systematically explore how device networking operates was very difficult.

One or more of the modular software elements encapsulate general mobile device functionality, such as: call control and handling; PIM functionality; SIM functionality; remote control, including screen scraping and faking key presses; monitoring, including processes, threads, memory and settings; UI, including creating an application where the screen elements are specified from a script; telephony, including monitoring and making calls; file system, including reading writing files and folders, monitoring for changes;

Rapid Dev

4

database, including structured storage, retrieval, searching and monitoring of arbitrary application data; device personalization, including ringtones, wallpaper and settings.

5 In one implementation, the element under the control of a high level interface is a TCP/IP interface which allows other programs on the device to be run upon receipt of an incoming connection or to make outgoing connections from the device under control of other device based programs. Another element under the control of a high level interface implements a remote command execution protocol. Another element under the control of a high level interface implements a scripting language that allows scripts to be written
10 which use other programs on the device also controlled by a command line interface.

A high level language program runs on an application development computer (such as a desktop PC) remote from the device that can send instructions to the or each element on the device controlled by the high level interface; the application development computer is
15 connected to the device over a local point to point IR, Bluetooth, USB, WAN, LAN, SMS or GPRS or any combination of these.

The high level language program is preferably a command line program that enables IP connections between the mobile device and a further program on the application development computer that implements the same remote command execution protocol
20 as the device. The high level language is not restricted to a single type of high level language, but can be any of the following depending on the requirements of the developer of the software application:

- (a) a command line interface;
- 25 (b) a scripting language;
- (c) a compiled language.

The high level language program can in addition run on the device, to enable re-programming of the device without the need to use a separate application development computer.

30 Rapid application development is then achieved by enabling device capabilities to be explored by executing the device-based elements controlled by a high level interface from a command prompt displayed on a display of the application development computer using the remote command execution protocol. An output of each command is shown

Rapid Dev

5

at the command prompt on the application development computer. Rapid application development is further achieved by using scripts which combine the results of several device-based elements controlled by a high level interface in the scripting language written on the device. The script can be composed in a text editor running on the application development computer. Finally, rapid application development is achieved by transferring the scripts to the device and executing them, again using the computer command prompt.

In an implementation (which will be described later in the Detailed Description section) the standard interface structure of a modular software element is the name of the element, a set of command line options, two input streams and two output streams. The modular software elements can be chained together to build complex functionality; they insulate the application developer from the specifics of the operating system of the device by requiring the application developer to understand the type of functionality to be deployed and not the specific operating specific code needed to implement that functionality using the operating system.

The device runs a command interpreter and the application development computer then runs a command execution shell.

In another implementation, the modular software elements execute on the device in the context of an identity and associated permissions. An identity server (located on the device) with secure permissions provides and controls the identity and associated permissions.

In one implementation, modular software elements, called pipe processors, are combined in a way that significantly reduces the time it takes to develop networked applications for mobile devices.

Pipe processors are stand alone modules that encapsulate a range of mobile device functionality. Pipe processors are written efficiently in a software code suitable for the phone operating system, such as C++. These pipe processors are all called from a standard interface structure, comprising the name of the pipe processor and a set of

Rapid Dev

6

options. The results of the pipe processor are returned to the calling element using a standard output and standard error.

Rapid networked application development is facilitated because:

5

- All of the pipe processors have the same type of interface that can be called from a command-line interface, or other high-level language. This provides the developer with the means of solving the network management problems of 1 but without having to learn the details of a specific network interface or program in a low-level language such as C++.
- All of the pipe processors can be executed on the device remotely from the PC, so providing the developer with a good user interface for development but without having to develop software first on an emulator and then for the device.
- The modular architecture of the pipe processors means that modules can be included or removed as necessary. This means that software can quickly be developed that also makes efficient use of the restricted resources on the mobile device, so solving problem 3. Other rapid development approaches for mobile devices, such as using high-level languages such as Visual Basic, require large run-time components and hence consume large resources on the mobile device.

10

15

20

The problem of rapid networked application development and reconfiguration has been around for some time as mobile devices, such as PDAs, have been around for many years. Current approaches to this problem, such as using Java MIDP, cannot however fully exploit the network features of the mobile device as they are constrained by the high-level interfaces required to make the development quick and easy. Also, many of the current approaches rely on the use of emulators on the PC. The present invention can complement Java MIDP to overcome these limitations. As noted earlier, the present invention hence solves the problem of rapid networked application development and reconfiguration, as all of the pipe processors can be called either from command-line, scripts, or other programming languages. Hence, required functionality can be quickly prototyped using scripting to prove the functionality, before being codified into a programming language for the application.

25

30

Rapid Dev

7

There are three significant further advantages to the present invention:

1. Enables programming by non-skilled programmers. Using the set of pipe processor components that can be called from both a command-line interface, scripting language and variety of programming languages enables both phone users with no programming experience to 'program' software on the phone as well as advanced software developers, all using the same components. This enables software to be modified by unskilled programmers to adapt it to uses that were not originally envisioned by a programmer, just by modifying the script on the phone. This can be used as a means of enabling non-skilled programmers to modify applications for their own use, or to quickly prototype and test an application that can be handed to a skilled developer for turning into a complete networked software application for mobile devices.
2. It allows someone to modify a software application when all they have is a mobile device, for example when they are on the train. Software can be developed from a PC, with a link to the mobile device. However, if the application is scripted on the mobile device then when you are away from your PC, the script can be quickly modified to create a different application, without the need to compilers, debuggers, emulators and the other development tools required for standard PC-based software development.
3. Provides a single interface from a wide range of programming languages, including command-line and scripting interfaces, to mobile devices running a wide range of operating systems. Hence, a programmer can choose whatever language they like to develop the software, and does not have to learn different interfaces for different mobile devices. This is similar in concept to using Java MIDP as a basis for writing portable applications for smartphones. However, using Java MIDP it is not possible to write good networked applications for mobile devices as Java MIDP standard does not allow access to the necessary networked features on the phone. This can be achieved by extending the MIDP programming interface with additional mobile-device specific interfaces, but this requires the developer to understand the different interfaces for each phone. The proposed framework eliminates this problem by providing a common interface to

Rapid Dev

8

the low-level networking and other phone features that is common across different mobile device operating systems.

5 In another aspect, there is software application developed using the method of rapid software application development described above. This software application may be initiated or controlled from a remote application development computer and may then be accessed or controlled by the remote application development computer in a secure fashion. The software application may also run stand-alone on the device without any initiation or control from a remote application development computer.

10

In terms of interaction with physical hardware, the modular software elements execute on the CPU of the mobile device. Further, because the elements execute using a command line interface, the interface necessarily has to be shown on a computer display.

15

Rapid Dev

47

CLAIMS

1. Method of rapid software application development for a wireless mobile device, the application being an enterprise networked application in which the device communicates with an enterprise server over one or more types of network connection; the method comprising the steps of:
- 5 (a) a developer using a standard, high level interface (that is not specific to the mobile device OS) on a computer remote from the mobile device to call, over one of the network connections, modular software elements running on the device, the modular elements each (i) encapsulating functionality required by the wireless mobile device and
- 10 (ii) executing on the device, under the control of an interpreter of the high level interface;
- (b) the developer causing elements on the device to be combined using a scripting engine running on the device; and
- (c) the developer exploring how different elements respond to inputs by
- 15 repeating steps (a) and (b).
2. The method of Claim 1 in which one or more modular elements encapsulate device networking functionality that relates to connectivity over one or more of the following: GPRS, 2G cellular, CDMA, WCDMA, Bluetooth, 802.11, infra-red, IP
- 20 networking, dial up, modem; HSCSD and EDGE.
3. The method of Claim 1 in which one or more of the modular software elements encapsulate general mobile device functionality.
- 25 4. The method of Claim 3 in which the general mobile device functionality relates to one or more of the following: call control and handling; PIM functionality; SIM functionality; remote control, including screen scraping and faking key presses; monitoring, including processes, threads, memory and settings;
- 30 UI, including creating an application where the screen elements are specified from a script; telephony, including monitoring and making calls; file system, including reading writing files and folders, monitoring for changes; database, including structured storage, retrieval, searching and monitoring of arbitrary application data; device personalization, including ringtones, wallpaper and settings.

Rapid Dev

48

5. The method of Claim 1 in which the element under the control of a high level interface is a TCP/IP interface which allows other programs on the device to be run upon receipt of an incoming connection or to make outgoing connections from the device under control of other device based programs.

5

6. The method of Claim 1 in which the element under the control of the high level interface implements a remote command execution protocol.

7. The method of Claim 1 in which the element under the control of the high level interface implements a scripting language that allows scripts to be written which use other programs on the device also controlled by a command line interface.

10

8. The method of Claim 1 in which a high level language program runs on an application development computer remote from the device that can send instructions to the or each element on the device controlled by the high level interface.

15

9. The method of Claim 8 in which the high level language program is a command line program that enables IP connections between the mobile device and a further program on the application development computer that implements the same remote command execution protocol as the device.

20

10. The method of Claim 9 in which rapid application development is achieved by enabling device capabilities to be explored by executing the device-based elements controlled by the high level interface from a command prompt of the application development computer using the remote command execution protocol.

25

11. The method of Claim 10 in which an output of each command is shown at the command prompt on the application development computer.

12. The method of Claim 9 in which rapid application development is achieved by using scripts which combine the results of several device-based elements controlled by a command line interface in the scripting language written on the device.

30

Rapid Dev

49

13. The method of Claim 12 in which the script is composed in a text editor running on the application development computer.

14. The method of Claim 12 or 13 in which rapid application development is achieved by transferring the scripts to the device and executing them, again using the computer command prompt.

15. The method of Claim 1 in which the standard interface of a modular software element is the name of the element, a set of command line options, two input streams and two output streams.

16. The method of Claim 8 in which the high level language is not restricted to a single type of high level language, but can be any of the following depending on the requirements of the developer of the software application:

- (a) a command line interface;
- (b) a scripting language;
- (c) a compiled language.

17. The method of Claim 16 in which the application development computer is a desktop PC.

18. The method of Claim 1 in which the high level language program can in addition run on the device, to enable re-programming of the device without the need to use a separate application development computer.

19. The method of Claim 1 in which the modular software elements insulate the application developer from the specifics of the operating system of the device by requiring the application developer to understand the type of functionality to be deployed and not the specific operating specific code needed to implement that functionality using the operating system.

20. The method of Claim 8 in which the device runs a command interpreter and the application development computer runs a command execution shell.

Rapid Dev

50

21. The method of Claim 8 in which the application development computer is connected to the device over a local point to point IR, Bluetooth, USB, WAN, LAN, SMS or GPRS or any combination of these.
- 5 22. The method of Claim 1 in which modular software elements can be chained together to build complex functionality.
23. The method of any preceding Claim in which the modular software elements execute on the device in the context of an identity and associated permissions.
- 10 24. The method of Claim 23 in which there is an identity server with secure permissions that provides and controls the identity and associated permissions.
25. The method of Claim 24 in which the identity server is located on the device.
- 15 26. The method of any preceding method claim in which the modular software elements execute on a CPU of the mobile device.
27. A software application developed using the method of any preceding Claim 1 -
- 20 26, the application comprising modular software elements, the modular elements each (i) encapsulating functionality required by a wireless mobile device and (ii) executing on the device, under the control of an interpreter of a high level interface.
28. The software application of Claim 27 which is initiated or controlled from a
- 25 remote application development computer.
29. The software application of Claim 28 which is accessed or controlled by the remote application development computer in a secure fashion.
- 30 30. The software application of Claim 27 which runs stand-alone on the device without any initiation or control from a remote application development computer.

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.